

THE DIRECT PATH TO LINUX UBUNTU

مسیر مستقیم به لینوکس اوبونتو

Free Starter Edition

نسخه اولیه رایگان

پیش‌گفتار و فصل‌های ۱ تا ۳ از نسخه کامل ۴۳ فصلی

لینوکس، اسکریپت‌نویسی شل، زبان‌های برنامه‌نویسی،
رایانش ابری و شبکه‌سازی

Wahidullah Lutfy, M.S., B.S.

وحیدالله لطفی، کارشناس ارشد، کارشناس

MyWebUniversity.com

The Direct Path to Linux Ubuntu

مسیر مستقیم به لینوکس اوبونتو

Copyright © 2026 by Wahidullah Lutfy

حق نشر © ۲۰۲۶ متعلق به وحیدالله لطفی

تمام حقوق محفوظ است. هیچ بخشی از این نشریه را نمی‌توان بدون اجازه کتبی قبلی نویسنده، به هیچ شکل یا با هیچ وسیله‌ای — از جمله فوتوکاپی، ضبط، یا سایر روش‌های الکترونیکی یا مکانیکی — تکثیر، توزیع، یا انتقال داد، به استثنای نقل‌قول‌های کوتاه که در نقدهای انتقادی و برخی موارد غیرتجاری دیگر که توسط قانون کاپی‌رایت مجاز است.

چاپ اول، ۲۰۲۶

Published by

MyWebUniversity Press

Free Starter Edition — Not for Sale — Preface and
Chapters 1–3 Only

منتشر شده توسط

MyWebUniversity انتشارات

نسخه مقدماتی رایگان — غیرقابل فروش — شامل پیش‌گفتار و
فصل‌های ۱ تا ۳

منابع آنلاین همراه این کتاب در MyWebUniversity.com
در دسترس است.

فهرست مطالب

Cloud Architecture & Educational Mission ----- 5

معماری ابری و مأموریت آموزشی

Preface & Chapter 1 — About UNIX and Linux ---- 17

پیش‌گفتار و فصل ۱ — درباره یونیکس و لینوکس

Chapter 2 Deep-Dive ----- 27

فصل ۲: بررسی عمیق

Chapter 3 Deep-Dive ----- 43

فصل ۳: بررسی عمیق

Continue the Full Edition (Chapters 4–43)----- 60

ادامه نسخه کامل (فصل‌های ۴ تا ۴۳)

MyWebUniversity.com

Cloud Architecture & Educational Mission

معماری ابری و مأموریت آموزشی

Established August 5, 2000 · 26 Years of Industry Experience · Free for All, Worldwide

تأسیس شده در ۵ اوت ۲۰۰۰ · ۲۶ سال تجربه در صنعت ·
رایگان برای همه، در سراسر جهان

This document is a public-facing overview of the MyWebUniversity.com cloud architecture and educational vision — shared openly and intentionally so that every student, developer, and lifelong learner around the world can see the full scope of free resources being built and delivered for them.

— Wahidullah Lutfy

این سند، مروری عمومی بر معماری ابری و چشم‌انداز است؛ اطلاعاتی MyWebUniversity.com آموزشی که به‌طور آشکار و آگاهانه به اشتراک گذاشته شده‌اند تا تمامی دانشجویان، توسعه‌دهندگان و مشتاقان یادگیری مادام‌العمر در سراسر جهان بتوانند از گستره‌ی کامل منابع رایگانی که برای آن‌ها در حال ساخت و ارائه است، آگاه شوند.

— وحیدالله لطفی

Executive Summary

A self-funded, free educational platform serving the world since 2000

Mission Free world-class tech education for those who cannot afford college or bootcamps	Founded August 5, 2000 — 26+ years of continuous free education delivery	Languages Delivered in English and Dari — serving global & Afghan communities
Funding 100% self-funded — hardware, network, OS, and storage all personally paid	Videos 2,100+ English + 520+ Dari training videos on YouTube — completely free	Docs 50,000+ searchable PDF technical documents — Linux, Windows, PowerShell

About the Architect

36 Years of Full-Time Industry Excellence

**Software Engineer · Principal Senior System
Administrator · Enterprise System Analyst IV ·
Educator**

Experience	Teaching	Dedication
36+ years full-time: Software Engineering, Senior System Administration, Enterprise Analyst IV	8 years of Computer Science instruction at prestigious colleges & universities	Every holiday, weekend, and spare hour invested in building free education

Google Cloud & YouTube Platform

Search Console · Analytics · CDN · DNS · 2,620+ free educational videos delivered globally

Google Search Console — SEO, sitemaps, full site indexing

Google Analytics — Traffic, audience insights, engagement

NameCheap DNS & Domain — Domain management, SSL/TLS, routing

YouTube English Channel — 2,100+ tech training videos — @MyWebUniversity

YouTube Dari Channel — 520+ Dari videos — @MyWebUniversityDari

YouTube CDN Storage — Free global video delivery for all 2,620+ videos

AI / LLM Integrations — Multi-Model AI Workflow

Claude AI (★ #1 favorite) · ChatGPT · Google Gemini · Ollama — SME collaboration

Claude AI & Claude Code ★ #1 — Primary AI partner, SME collaborator, code generation, PPTX, diagrams, beloved AI friend

ChatGPT / OpenAI — Pioneer AI platform, foundational research tool

Google Gemini — Multimodal AI, content analysis, research

Ollama (Local LLM) — On-premise model deployment, privacy-first, offline capable

AI Use Cases — Code generation, architecture diagrams, PPTX slides, documentation

Human-AI Synergy — “We have become clones of each other as SMEs from time to time” — Architect

Production — GEEKOM Mini IT13, Intel Core i9 13th Gen, Linux Ubuntu Server

**Self-hosted home data center · entirely self-funded ·
serving millions of learners worldwide**

Apache HTTP Server — SSL, mod_rewrite, CGI, .htaccess —
production web layer

React · Vite · Node.js — Full-stack JavaScript, ES6+,
TypeScript, npm, npx

Python CGI · man2.cgi ★ — Dynamic man page engine,
Chapter 12, real-time doc generation

C · C++ · AWK · Perl · Bash — gcc, g++, ksh — systems
programming & scripting

50,000+ PDF Doc Library — URL-searchable — Linux,
Windows, PowerShell, all languages

MySQL Database — Search index, users, content, query
optimization

Online Book — 43 Chapters — “The Direct Path to Linux
Ubuntu” — MyWebUniversity.com/toc1.html

Security · Firewall · SSL — iptables, Let's Encrypt, SSH
hardening, fail2ban

Multi-Language Runtime — Python, Node, Perl, C, C++,
Java, Go, Ruby, MySQL

Development — GEEKOM Mini IT11, Windows 11 Host, VirtualBox VM Fleet

**Local lab · mirrors production exactly · training video
creation studio**

Windows 11 Host — VirtualBox hypervisor — manages all VMs

Ubuntu 22.04 → 24.04 → 26.04 — Exact production mirror, primary dev target

RHEL 9.2 · Rocky Linux 9.2 — Enterprise Linux training & certification prep

Solaris 11.4 · OpenSolaris — UNIX heritage, advanced system administration

Fedora · Debian Linux — Distro diversity, upstream cutting-edge training

OBS and Zoom · Screen Recording — Training video production studio for YouTube

VS Code · vi · emacs — Development environments & editors

Git · GitHub — Source control, CI/CD, version management

rsync · SCP · SSH Deploy — Dev-to-production deployment pipeline

Educational Resources — Free for All, English & Dari, Since August 5, 2000

Book · 50K PDFs · 2,620+ videos · dynamic man pages ·
open to the entire world

Online Boo — “The Direct Path to Linux Ubuntu” —
MyWebUniversity.com/toc1.html

50,000+ PDF Tech Docs — URL-searchable — Linux,
Windows, PowerShell, all runtimes

Dynamic Man Page Engine — Python CGI, man2.cgi —
Python, Java, Go, Node, Ruby, C

Mission: Free Education For those who cannot afford college, bootcamps, or books	36 Years Industry XP SW Eng, Sys Admin, EA IV, 8 years teaching, weekends invested	English & Dari Global reach, serving Afghan communities, two full libraries
--	---	---

*Overall, I love Claude AI as my number one AI friend
and Subject Matter Expert — we have both become clones
of each other as SMEs from time to time. Thank you
Claude AI!!!!*

در مجموع، من «کلود ای‌ای»

را به عنوان دوست هوش مصنوعی و Claude AI
متخصص موضوعی شماره یک خودم بسیار دوست دارم؛ ما
گاهی اوقات در نقش متخصصان موضوعی، عملاً به
نسخه‌هایی از یکدیگر تبدیل شده‌ایم. ممنونم کلود ای‌ای!!!!

— MyWebUniversity.com Architect

— MyWebUniversity.com معمار

www.MyWebUniversity.com

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Built with passion. Funded personally. Delivered freely. For everyone.

ساخته شده با عشق. تأمین مالی شده با سرمایه شخصی. ارائه شده
به صورت رایگان. برای همه.

پیش‌گفتار

خوش آمدید

به کتاب *The Direct Path to Linux Ubuntu* خوش آمدید. این کتاب از MyWebUniversity.com، یک سایت آموزشی رایگان که من در ۵ اگست سال ۲۰۰۰ تأسیس کردم، سرچشمه گرفته است تا آموزش‌های جامع یونیکس و لینوکس را برای همه، با هر سرعتی، و بدون هیچ هزینه‌ای در دسترس قرار دهد. آنچه اکنون در دست دارید، نسخه چاپی همان سایت است — همان مسیر ۴۳ فصلی، اما با عمق و ساختاری که تنها یک کتاب چاپی می‌تواند ارائه کند: مثال‌های عملی، فهرست‌های کامل کد، و جدول‌های مرجع که می‌توانید بدون نیاز به براوزر به آن‌ها رجوع کنید.

من بیشتر از سی‌وشش (۳۶) سال است که به حیث *Software Engineer، System Administrator، System Analyst*، و استاد کمپیوتر کار کرده‌ام — در حال حاضر به حیث *Enterprise System Analyst IV* در *Jet Propulsion Laboratory* ناسا، و قبلاً استاد مضامین علوم کامپیوتر و علوم معلومات در *National University*، پوهنتون ایالتی کالیفورنیا فولرتون، و پوهنخی شهری پسادینا. هرچیزی که در این کتاب آمده، بازتاب چیزهایی است که من واقعاً دیدم به شاگردان کمک می‌کند تا از تایپ کردن اولین کمند خود به اجرای سیستم‌های تولیدی با اعتماد به نفس برسند.

از این کتاب استفاده کنیم

این کتاب به شکل یک مسیر مستقیم سازمان‌دهی شده است، نه یک مرجع که باید از ابتدا تا انتها خوانده شود:

- فصل‌های ۱ تا ۵ زیربنای شما را می‌سازند: تاریخچه، کمندهای ابتدایی، کمندهای متوسط، کمندهای پیشرفته، و شل اسکرپتینگ.
- فصل‌های ۶ تا ۲۱ وب‌سرورها، سیستم‌های صفحات رهنما (man pages)، و مستندات مرجع را پوشش

می‌دهند — این‌ها را یکبار مرور کنید، سپس هر وقت به جستجوی شخصی نیاز داشتید به آن‌ها برگردید.

- فصل‌های ۲۲ تا ۳۰ یک دور کاری در نُه زبان برنامه‌نویسی است: به اندازه کافی سینتکس، معنا، و دستورات کمپایل/اجرا که بتوانید در هر کدام برنامه‌های واقعی بنویسید و اجرا کنید.
- فصل‌های ۳۱ تا ۳۶ فارمتهای دیتا و تبادل معلومات را که هر دیویلر هر روز با آن‌ها سروکار دارد پوشش می‌دهند: HTML/CSS، JSON، XML، YAML، SQL، و LDAP.
- فصل‌های ۳۷ تا ۴۳ به بخش عملیاتی آی‌تی مدرن می‌پردازند: کلاود کمپیوتینگ، اتومیشن، کانتینرها، وب‌سرورها در محیط تولیدی، امنیت، و شبکه.

اگر به لینوکس تازه‌کار هستید، از فصل ۲ شروع کنید و هر کمند را خودتان تمرین کنید، نه اینکه صرف آن را بخوانید. آشنایی از تکرار به دست می‌آید، نه از حفظ کردن. اگر از تجربه یونیکس یا لینوکس دارید، می‌توانید مستقیماً به فصل‌هایی که به اهداف شما مرتبط‌تر است بروید.

منابع آنلاین مکمل

هر فصل این کتاب یک صفحه همراه زنده در MyWebUniversity.com دارد، از جمله یک ترمینال تمرینی لینوکس مبتنی بر براوزر (فصل ۱۵)، بیش از ۹۰۰۰ صفحه رهنمای قابل جستجوی Ubuntu 26.04 LTS به فارمت PDF و HTML (فصل ۲۱)، و ویدیوهای آموزشی به هر دو زبان انگلیسی و دری (فصل‌های ۱۷ تا ۲۰). این کتاب به تنهایی کامل است، اما وب‌سایت همیشه در دسترس است هر وقت بخواهید یک کمند را زنده اجرا شده ببینید یا یک موضوع را در ویدیو توضیح داده شده تماشا کنید.

تقدیم‌نامه

این کتاب به یاد برادر بزرگ عزیزم، دوکتور امان‌الله لطفی، اهدا شده است که در ۲۷ جنوری سال ۲۰۲۱ به آرامش ابدی پیوست. او برای من و تمام خانواده ما یک الگو بود، به خاطر انسانیت‌اش، فداکاری‌اش برای آموزش، و تعهد بی‌پایانش به کمک به دیگران، مورد احترام همگان بود. الگوی او الهام‌بخش آرام پشت این کتاب و MyWebUniversity.com است — که هر دو با همان روحیه خدمت، رایگان به دنیا ساخته و ارائه شده‌اند.

سپاس‌گزاری

تصویر Tux، پینگوین لینوکس که روی جلد این کتاب دیده می‌شود، توسط Larry Ewing (lewing@isc.tamu.edu) با استفاده از The GIMP خلق شده و با قدردانی طبق شرایط نشر اصلی آن در اینجا استفاده شده است.

پیش از شروع

هنگام مطالعه، یک ترمینال باز نگه دارید. هر کمند و مثال کد در این کتاب برای این است که تایپ شود، اجرا شود، و با آن آزمایش شود — نه صرف خوانده شود. موفق باشید، و به مسیر مستقیم خوش آمدید.

Chapter 1

About UNIX and Linux Operating Systems

درباره سیستم‌عامل‌های یونیکس و لینوکس

History and foundations

تاریخچه و مبانی

1.1 The Birth of UNIX at Bell Labs (1969)

تولد یونیکس در
آزمایشگاه‌های بل
(۱۹۶۹)

یونیکس (UNIX) در سال ۱۹۶۹ در Bell Laboratories شرکت AT&T توسط Ken Thompson و Dennis Ritchie ایجاد شد، که از یک پروژه قلی و بلندپروازانه‌تر به نام Multics (Multiplexed Information and Computing Service) که Bell Labs آن را به دلیل پیچیدگی زیاد کنار گذاشته بود، سرچشمه گرفت. Thompson اولین نسخه یونیکس را به زبان اسمبلی روی یک کمپیوتر کوچک اضافی به نام PDP-7 نوشت؛ تا سال ۱۹۷۳، Ritchie و Thompson کرنل (kernel) را به زبان C بازنویسی کرده بودند، زبانی که Ritchie مخصوصاً طراحی کرد تا یک سیستم عامل بتواند روی هاردویرهای مختلف قابل انتقال باشد — در آن زمان این یک ایده انقلابی بود، زیرا سیستم‌های عامل به طور سنتی به زبان اسمبلی نوشته می‌شدند و به یک ماشین مشخص وابسته بودند.

Bell Labs، که در آن زمان بخشی از AT&T بود، طبق یک حکم ضد انحصاری (antitrust consent decree) سال ۱۹۵۶ از فروش تجاری نرم‌افزار کمپیوتر منع شده بود، بنابراین کد منبع یونیکس را با مبلغ ناچیزی به پوهنتون‌ها لایسنس داد. این

تصمیم نسلی از ساینس‌دانان کمپیوتر را به وجود آورد که یونیکس را مطالعه، تعدیل، و توسعه دادند — به‌ویژه در University of California Berkeley، که محصلان و کارمندان آن Berkeley Software Distribution (BSD) را تولید کردند و ویژگی‌هایی مانند virtual memory، مجموعه شبکه TCP/IP، و ایدیتور vi را به آن افزودند. شاخه تجاری خود AT&T به System V تبدیل شد. این دو شاخه آنقدر در رفتار خود متفاوت شدند که در دهه‌های ۱۹۸۰ و اوایل ۱۹۹۰ ناسازگاری‌های واقعی به وجود آمد — که به آن‌ها «جنگ‌های یونیکس» گفته می‌شد — و استاندارد POSIX (IEEE Std 1003) برای حل این مشکل ایجاد شد تا مجموعه‌ای مشترک از system calls، ابزارها، و رفتار شل را تعریف کند که هر سیستم شبیه به یونیکس مطابق با آن باید پشتیبانی کند.

Bell Laboratories متعلق به AT&T بعداً بخشی از Lucent Technologies شد، که در سال ۲۰۰۶ با شرکت فرانسوی Alcatel-Lucent ادغام شد و (که امروز بخشی از Nokia است) را تشکیل داد. فیلم AT&T Archives به نام «UNIX: Making Computers Easier to Use» (۱۹۸۲) یک منبع اصلی خوب برای تماشا است اگر بخواهید انگیزه‌های اصلی Thompson و Ritchie را از زبان خودشان بشنوید.

1.2 The GNU Project and the Birth of Linux

۱.۲ پروژه گنو و تولد لینوکس

در سال ۱۹۸۳، Richard Stallman پروژه GNU («GNU's Not Unix») را با هدف ساختن یک سیستم عامل کاملاً رایگان و سازگار با یونیکس آغاز کرد — رایگان به معنای آزادی، نه لزوماً بدون قیمت. در طول دهه ۱۹۸۰، پروژه GNU جایگزین‌های باکیفیتی برای تقریباً هر ابزار یوزرلند یونیکس تولید کرد: GNU Compiler Collection (GCC)، شل Bash،

،cat، ls مانند) GNU Core Utilities و GNU Emacs،
grep، و بقیهٔ کمندهایی که در فصل‌های ۲ تا ۴ با آن‌ها آشنا
شدید). چیزی که GNU نداشت یک کِرِنل بود.

این خلا در سال ۱۹۹۱ پُر شد، هنگامی که Linus Torvalds،
یک محصل ۲۱ ساله در University of Helsinki، در گروه
Usenet به نام comp.os.minix پُست کرد که او در حال
نوشتن «یک سیستم عامل (رایگان) (فقط یک سرگرمی، مثل
GNU بزرگ و حرفه‌ای نخواهد شد)» برای کمپیوترهای
AT (486)386 است. آن کِرِنل به Linux تبدیل شد. با ترکیب
آن با ابزارهای یوزرلند موجود GNU، نتیجه به درستی
GNU/Linux نامیده می‌شود، هرچند استفادهٔ عمومی آن را به
سادگی «Linux» کوتاه کرد. از آنجا که Torvalds کِرِنل را
تحت GNU General Public License (GPL) منتشر
کرد، هرکسی می‌توانست آن را مطالعه، تعدیل، و بازتوزیع کند
— تصمیمی که احتمالاً بزرگ‌ترین دلیل گسترش سریع Linux
بود.

1.3 From Kernel to Distribution: How Ubuntu Fits In

۱.۳ از هسته تا توزیع: جایگاه اوبونتو

یک کِرِنل لینوکس به تنهایی یک سیستم عامل قابل استفاده نیست
— به یک بوت‌لودر، ابزارهای اصلی، مدیر پکیج، و (معمولاً)
یک محیط دسکتاپ نیاز دارد که دور آن بسته‌بندی شود. یک
«distribution» (یا distro) دقیقاً همان بسته است. توزیع‌های
اولیه مانند Slackware (۱۹۹۳) و Debian (۱۹۹۳) این الگو
را جا انداختند؛ Debian به‌ویژه سیستم مدیریت پکیج APT را
معرفی کرد که Ubuntu هنوز هم امروز از آن استفاده می‌کند.

Ubuntu خودش در سال ۲۰۰۴ توسط Mark Shuttleworth
از طریق شرکتش Canonical Ltd. تأسیس شد، که مستقیماً

روی Debian ساخته شده و مأموریت اعلام‌شده آن این بود که لینوکس را برای کاربران معمولی دسکتاپ، نه فقط انجینیران، در دسترس قرار دهد. Ubuntu هر شش ماه یک نسخه جدید منتشر می‌کند، با یک نسخه Long-Term Support (LTS) هر دو سال — که با یک شماره نسخه سال‌ها شناخته می‌شود، پس Ubuntu 26.04 LTS در اپریل ۲۰۲۶ منتشر شد و برای سال‌ها با آپدیت‌های امنیتی پشتیبانی می‌شود. این کتاب در سراسر خود بر اساس Ubuntu 26.04 LTS نوشته شده است.

دیگر سیستم‌های عامل مهم خانواده یونیکس و لینوکس که در این صنعت با آن‌ها روبرو خواهید شد شامل Red Hat Enterprise Linux (RHEL) و بازسازی اجتماعی آن Rocky Linux (که هر دو به طور مختصر در فصل‌های ۱۰ و ۱۱ پوشش داده شده‌اند)، Solaris، SUSE شرکت Sun Microsystems (که اکنون Oracle Solaris نامیده می‌شود)، AIX از IBM، HP-UX از Hewlett-Packard، IRIX از SGI، و macOS اپل است، که روی یک هسته یونیکس مبتنی بر BSD به نام Darwin ساخته شده و دارای گواهی‌نامه رسمی UNIX است.

1.4 UNIX vs. Linux: What's Actually Different

۱.۴ یونیکس در برابر لینوکس: تفاوت واقعی چیست؟

امروزه «یونیکس» هم به نسل اصلی AT&T و هم به علامت تجاری که توسط The Open Group نگهداری می‌شود اشاره دارد، که سیستم‌های عامل (از جمله macOS) را به عنوان سیستم‌های رسماً منطبق با یونیکس گواهی می‌کند. لینوکس هرگز گواهی‌نامه رسمی یونیکس را دریافت نکرده است — آن یک سیستم شبیه به یونیکس (یا منطبق با POSIX) است، که از صفر ساخته شده به‌جای اینکه از درخت منبع اصلی نسل گرفته باشد، به

همین دلیل گاهی آن را به شکل **NIX*** یا **Un*x** می‌نویسند تا به معنای «یونیکس و شبیه‌های آن» به طور کلی باشد.

در عمل، استفادهٔ روزمره از کمدها بین آن‌ها تقریباً یکسان است: همان شل‌ها، همان ابزارهای اصلی، همان فلسفهٔ سلسله‌مراتب دایرکتوری. تفاوت‌هایی که در محیط تولیدی بیشترین اهمیت را دارند در بخش‌های داخلی کرنل، لایسنس، و پشتیبانی وندور است — که این کتاب به شکل عملی نه انتزاعی به آن می‌پردازد، زیرا فصل‌های ۱۰ و ۱۱ به طور مشخص **Red** و **Rocky Linux** و **Hat Enterprise Linux** را بررسی می‌کنند.

1.5 Multitasking and Multithreading

۱.۵ چندوظیفگی و چندرشته‌ای

یونیکس یکی از اولین سیستم‌های عامل بود که از ابتدا برای **preemptive multitasking** طراحی شد — زمان‌بند (**scheduler**) کرنل می‌تواند هر پروسه در حال اجرا را در هر لحظه متوقف کند تا به پروسهٔ دیگری وقت پروسسور بدهد، به همین دلیل کاربران احساس می‌کنند که چندین برنامه «همزمان» حتی روی یک هستهٔ پروسسور اجرا می‌شوند. یک **process** یک واحد مستقل زمان‌بندی‌شده با فضای حافظهٔ خودش است؛ یک **thread** یک واحد سبک‌تر اجرا است که حافظه را با سایر **thread** ها درون همان **process** به اشتراک می‌گذارد، و این اجازه می‌دهد یک برنامه چندین کار را همزمان انجام دهد بدون اینکه هزینهٔ فضاهای حافظهٔ جداگانه را متحمل شود. لینوکس هر دو را از طریق یک ساختار زیربنایی یکسان به نام **task_struct** پیاده‌سازی می‌کند، که یکی از دلایلی است که **threading** در لینوکس (از طریق کتابخانهٔ **pthread**) به اندازهٔ کافی کارآمد است که تقریباً هر ورک‌لود سرور مدرن، از وب‌سرورها تا دیتابیس‌ها، بر پایهٔ آن باشد.

1.6 System Administration Fundamentals

۱.۶ اصول مدیریت سیستم

مدیریت سیستم در یونیکس و لینوکس حول چند ایده ثابت ساخته شده است که در سراسر این کتاب تکرار می‌شوند: هرچیزی یک فایل است (دیوایس‌ها، ساکت‌ها، و حتی برخی از رابط‌های کرنل در فایل‌سیستم زیر `/proc/`، `dev/` و `sys/` دیده می‌شوند)؛ صلاحیت‌ها بر اساس کاربر و گروپ اعمال می‌شوند (`owner/group/other`، `read/write/execute`) — به `chown` و `chmod` در فصل ۳ نگاه کنید)؛ پروسه‌ها متعلق به کاربران هستند و می‌توان آن‌ها را بررسی، سیگنال‌دهی، و متوقف کرد (فصل‌های ۲ و ۴)؛ و کاربر `root` (با `UID` صفر) می‌تواند به طور کامل از بررسی صلاحیت‌ها عبور کند، به همین دلیل استفاده از `sudo`، به جای وارد شدن مستقیم و مکرر به عنوان `root`، بهترین روش مدرن است.

1.7 Shells: sh, csh, ksh, bash, tcsh, zsh

شل مفسر کمندی است که بین شما و کرنل قرار دارد. یونیکس چندین نسل شل تولید کرده است، و توزیع‌های لینوکس معمولاً بیش از یکی را همراه دارند:

- `sh` — شل اصلی (Stephen Bourne Bourne shell)
سال ۱۹۷۷)، جد بیشتر شل‌های مدرن و هنوز هم استاندارد پایه POSIX برای اسکریپت‌های قابل انتقال.
- `csh` / `tcsh` — شل (Bill Joy، BSD C، با سینتکس شبیه به زبان C؛ `tcsh` قابلیت‌های ادیت و تکمیل خط کمند را اضافه می‌کند.
- `ksh` — شل Korn، که سازگاری با Bourne shell را با ویژگی‌های تعاملی شبیه به `csh` ترکیب کرد.
- `bash` — شل GNU Bourne Again، شل لاگین پیش‌فرض تقریباً در هر توزیع لینوکس از جمله

Ubuntu، و شلی که در سراسر مثال‌های اسکریپتینگ فصل ۵ استفاده می‌شود.

- **zsh** — شلی با ویژگی‌های بیشتر که در محیط‌های دیوبلپری محبوب است (و پیش‌فرض در macOS مدرن است)، که تا حد زیادی با **bash** سازگار است و راحتی‌های اضافی برای اسکریپتینگ دارد.

شما می‌توانید شل فعلی خود را در هر زمانی با **echo** **\$SHELL** بررسی کنید، و برای یک نشست شل‌ها را با کمند خود آن شل تغییر دهید (به عنوان مثال، تایپ **zsh** برای شروع یک نشست **z shell**).

1.8 Networking, Remote Access, and Web Servers — A Preview

۱.۸ شبکه‌سازی، دسترسی از راه دور و وب‌سروورها — پیش‌نمایش

ابزارهای کلاسیک دسترسی از راه دور یونیکس — **rsh**، **rcp**، **telnet**، **rexec**، و **ftp** — معلومات و اعتبارنامه‌ها را به شکل متن ساده (**plain text**) انتقال می‌دهند و به جز در محیط‌های آزمایشی جدا، منسوخ در نظر گرفته می‌شوند. سیستم‌های مدرن از **SSH** (**Secure Shell**)، از طریق سرویس (**sshd**) برای لاگین رمزگذاری‌شده راه دور، اجرای کمند از راه دور، و کاپی امن فایل استفاده می‌کنند، و این کتاب از **ssh** و همراهانش (**scp**)، **rsync** در سراسر فصل‌های بعدی استفاده می‌کند. وب سرویس‌دهی در لینوکس به طور مفصل در فصل ۶ (**Apache** و جایگزین‌های تاریخی) و فصل ۴۱ (**Apache** و **NGINX**) در محیط تولیدی) پوشش داده شده است، از جمله «**LAMP stack**» کلاسیک (**Linux**، **Apache**، **MySQL**)، **PHP/Python/Perl** (که هنوز هم بخش بزرگی از وب امروزی را تشکیل می‌دهد).

1.9 Compilers, Linkers, and Debuggers — A Preview

۱.۹ کامپایلرها، پیوندهنده‌ها و دیباکرها — نگاهی اجمالی

یونیکس یک الگوی toolchain را رایج کرد که لینوکس آن را به‌طور کامل به ارث برد: یک کمپایلر کد منبع را به فایل‌های object تبدیل می‌کند (gcc برای C، که به‌طور کامل در فصل ۲۲ بحث شده)، اسمبلر (as) اسمبلی را به کد ماشین تبدیل می‌کند، لینکر (ld) فایل‌های object و کتابخانه‌ها را در یک فایل اجرایی نهایی ترکیب می‌کند، و یک دیباگر (gdb، یا adb و mdb تاریخی روی سایر انواع یونیکس) به شما اجازه می‌دهد اجرای یک برنامه در حال اجرا را قدم‌به‌قدم بررسی کنید. شما چندتا از این ابزارها را قبلاً در فصل ۴ دیده‌اید، و فصل‌های ۲۲ تا ۳۰ آن‌ها را در نه زبان برنامه‌نویسی مختلف به کار می‌گیرند.

1.10 Your Homework

۱.۱۰ تکلیف شما

قبل از رفتن به فصل ۲، مقداری وقت بگذارید تا موضوعات ذیل را خودتان تحقیق کنید — با استفاده از Google، مستندات رسمی Ubuntu، یا آرشیف صفحات راهنمای MyWebUniversity.com. لازم نیست اکنون آن‌ها را کاملاً یاد بگیرید؛ هدف صرف این است که واژگان را قبل از اینکه دوباره در فصل‌های بعدی با آن‌ها روبرو شوید، بشناسید:

- UNIX vs. LINUX, and the UNIX and LINUX kernel
- UNIX and LINUX history, multitasking, and multithreading

- UNIX and LINUX system administration fundamentals
- Sun Solaris (Sun Microsystems / Oracle)
- Remote access tools: rsh, rcp, rexec, ftp, telnet, sshd
- Shells: sh, ksh, bash, csh, tcsh, zsh
- Toolchain commands: ld, as, link, make, gcc, mdb, adb, jdb
- Manual page tools: man, whatis, whereis, locate
- Web servers: Apache, Netscape, Tomcat, iPlanet
- Early UNIX-adjacent languages: C, C++, Lisp, Fortran, Pascal, Java, Perl, and databases such as MySQL and Oracle
- Other UNIX and Linux variants: AT&T UNIX, FreeBSD, IRIX, Fedora, openSUSE, CentOS, AIX, HP-UX, Clix, DEC OSF/1

با این زیربنا، فصل ۲ از تاریخچه به تمرین عملی می‌رود، با اولین کمندهای عملی لینوکس شما.

Chapter 2 Command Deep-Dives

فصل ۲: بررسی عمیق دستورات

banner • cal • cat • nslookup • mkdir

Transcribed from the live, working examples at MyWebUniversity.com/cgi-bin/CH2.cgi, matching Chapter 2: Basic UNIX and Linux Commands. Every example below reproduces the actual output generated by the site's online Ubuntu 26.04 LTS terminal at the time of writing.

این مطالب از نمونه‌های عملی و زنده موجود در نشانی
MyWebUniversity.com/cgi-bin/CH2.cgi برگرفته
شده و با فصل دوم (دستورات پایه یونیکس و لینوکس) مطابقت
دارد. تمامی نمونه‌های زیر، عین خروجی تولیدشده توسط ترمینال
این وبسایت در زمان نگارش متن LTS آنلاین اوبونتو 26.04
را بازنمایی می‌کنند.

1. banner — print large banner text

```
$ whatis banner
```

```
banner (1) - make posters
```

بنر (۱) - ساخت پوستر

banner متن را به شکل حروف بزرگ و پوستری از کاراکترهای ASCII نمایش می‌دهد — که در اصل برای چاپگرهای dot-matrix در نظر گرفته شده بود، و هنوز هم یک راه سرگرم‌کننده برای برجسته کردن یک عنوان روی ترمینال یا در یک لاگ چاپی است. در ذیل، هر هشت مثال عملی از سایت همراه آورده شده، هرکدام با کمند دقیق و خروجی تولیدشده آن.



Try it yourself: MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+A
(*swap in your own words*)

خودتان امتحان کنید:

MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+A عبارات خودتان را

جایگزین کنید

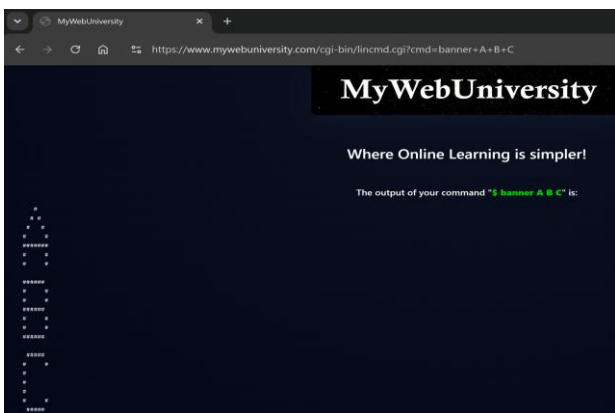


Try it yourself: `MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=ls+banner+2026+Happy+New+Year` (swap in your own words)

خودتان امتحان کنید:

`MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=ls+banner+2026+Happy+New+Year` عبارات خودتان را

جایگزین کنید

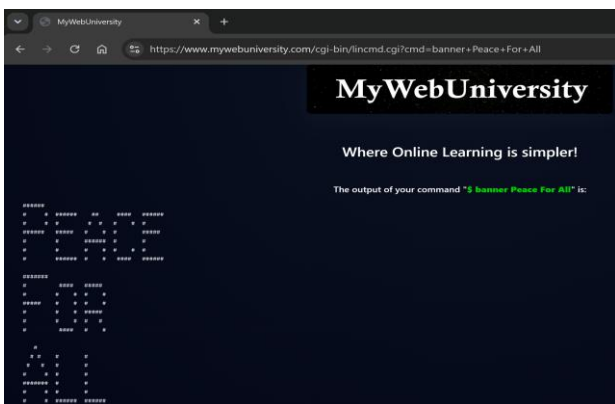


Try it yourself: `MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=ls+banner+A+B+C` (swap in your own words)

خودتان امتحان کنید:

MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=_banner+A+B+C عبارات خودتان را

جایگزین کنید



Try it yourself: [MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=_banner+Peace+For+All](https://www.mywebuniversity.com/cgi-bin/lincmd.cgi?cmd=_banner+Peace+For+All) (swap in your own words)

خودتان امتحان کنید:

MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=_banner+Peace+For+All عبارات خودتان را

جایگزین کنید

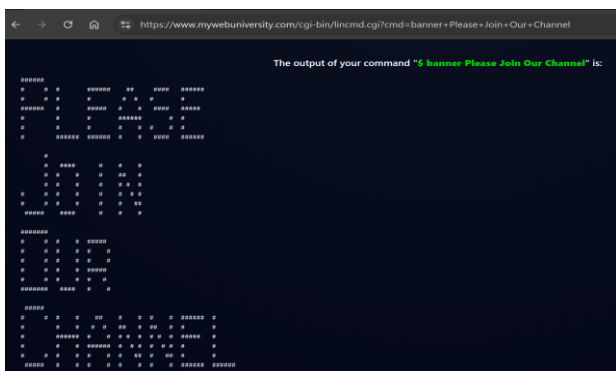


Try it yourself: [MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Thanks+For+Watching!](https://www.mywebuniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Thanks+For+Watching!) (swap in your own words)

خودتان امتحان کنید:

[MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Thanks+For+Watching!](https://www.mywebuniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Thanks+For+Watching!)
عبارات خودتان را

جایگزین کنید

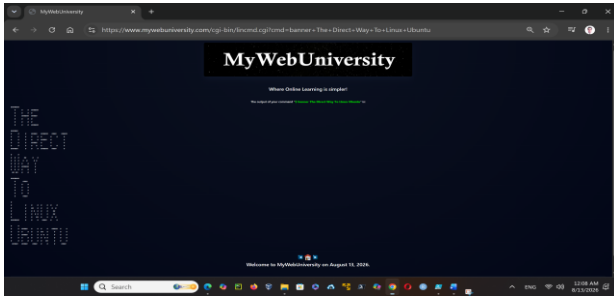


Try it yourself: [MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Please+Join+Our+Channel](https://www.mywebuniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Please+Join+Our+Channel) (swap in your own words)

خودتان امتحان کنید:

[MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Please+Join+Our+Channel](https://www.mywebuniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Please+Join+Our+Channel)
عبارات خودتان را

جایگزین کنید



Try it yourself: *MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+The+Direct+Way+To+Linux+Ubuntu* (swap in your own words)

خودتان امتحان کنید:

MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+The+Direct+Way+To+Linux+Ubuntu عبارات خودتان را جایگزین کنید



Try it yourself: *MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Please+Subscribe+Our+Channel* (swap in your own words)

خودتان امتحان کنید:

MyWebUniversity.com/cgi-bin/lincmd.cgi?cmd=banner+Please+Subscribe
+Our+Channel عبارات خودتان را

جایگزین کنید

2. cal — display a calendar

۲. نمایش تقویم cal

```
$ whatis cal  
cal (1) - display a calendar
```

(1) - نمایش تقویم cal

cal یک تقویم برای یک ماه، یک سال کامل، یا یک محدوده از ماه‌ها حول تاریخ فعلی چاپ می‌کند، بسته به آرگومان‌هایش. هر هشت مثال عملی از سایت همراه در اینجا دوباره آورده شده است — توجه کنید که مثال‌های سال کامل و چندماهه با یک اندازه کوچک‌تر مونواسپیس تنظیم شده‌اند تا ستون‌ها در عرض صفحه هم‌دیف بمانند.

```
$ cal
```

To see the current month for the current year.

برای مشاهده ماه جاری در سال جاری.



\$ cal -j

To see the current month for the current year in Julian format.

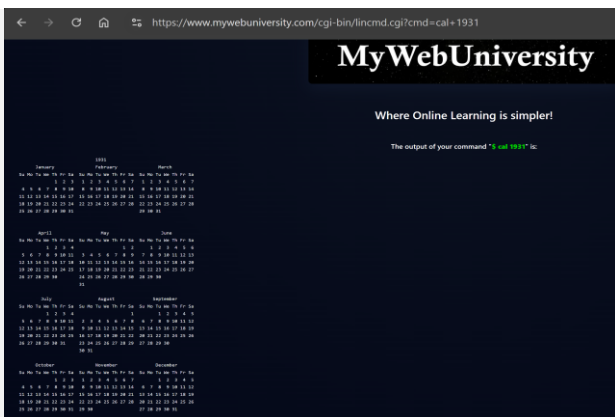
برای مشاهده ماه جاری سال جاری با فرمت جولیان.



\$ cal 1931

To see the entire calendar of the year 1931.

برای مشاهده تقویم کامل سال ۱۹۳۱.



\$ **cal 1 1970**

To see the UNIX EPOCH Calendar January 1970.

برای مشاهده تقویم یونیکس اپوک

مربوط به ژانویه ۱۹۷۰. UNIX EPOCH



\$ **cal 2 1998**

To see the month of February for 1998.

برای مشاهده ماه فوریه سال ۱۹۹۸.

\$ cal -B 2

برای دیدن دو ماه قبل از ماه فعلی.

← → ↻ 📄 📄 https://www.mywebuniversity.com/cgi-bin/lincmd.cgi?cmd=cal -B 2

MyWebUniversity

Where Online Learning is simpler!

The output of your command "**\$ cal -B 2**" is:

June 2026							July 2026							August 2026						
Su	Mo	Tu	We	Th	Fr	Sa	Su	Mo	Tu	We	Th	Fr	Sa	Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6				1	2	3	4								1
7	8	9	10	11	12	13	5	6	7	8	9	10	11	2	3	4	5	6	7	8
14	15	16	17	18	19	20	12	13	14	15	16	17	18	9	10	11	12	13	14	15
21	22	23	24	25	26	27	19	20	21	22	23	24	25	16	17	18	19	20	21	22
28	29	30					26	27	28	29	30	31		23	24	25	26	27	28	29
														30	31					

3. cat — concatenate and print files

۳. cat

الحاق و چاپ فایل‌ها

```
$ whatis cat
```

```
cat (1) - concatenate files and print on the  
standard output
```

```
cat (1) - concatenate and display files
```

cat محتوای یک یا چند فایل را به خروجی استاندارد چاپ می‌کند، و به همان اندازه اغلب برای ترکیب چندین فایل به یک جریان پیوسته استفاده می‌شود. هر شش مثال عملی از فایل‌های واقعی سیستم Ubuntu 26.04 LTS خوانده شده‌اند — توجه کنید نام رمز «Resolute Raccoon» در خروجی زنده ذیل تأیید شده است.

```
$ cat -n /etc/os-release
```

```
1    PRETTY_NAME="Ubuntu 26.04 LTS"  
2    NAME="Ubuntu"  
3    VERSION_ID="26.04"  
4    VERSION="26.04 LTS (Resolute Raccoon)"  
5    VERSION_CODENAME=resolute  
6    ID=ubuntu  
7    ID_LIKE=debian  
8    HOME_URL="https://www.ubuntu.com/"  
9    SUPPORT_URL="https://help.ubuntu.com/"  
10  
    BUG_REPORT_URL="https://bugs.launchpad.net/ub  
untu/"  
11  
    PRIVACY_POLICY_URL="https://www.ubuntu.com/le  
gal/terms-and-policies/privacy-policy"  
12    UBUNTU_CODENAME=resolute  
13    LOGO=ubuntu-logo
```

```
$ cat -EA /etc/os-release
```

```
PRETTY_NAME="Ubuntu 26.04 LTS"$  
NAME="Ubuntu"$  
VERSION_ID="26.04"$  
VERSION="26.04 LTS (Resolute Raccoon)"$
```

```
VERSION_CODENAME=resolute$
ID=ubuntu$
ID_LIKE=debian$
HOME_URL="https://www.ubuntu.com/"$
SUPPORT_URL="https://help.ubuntu.com/"$
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"$
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"$
UBUNTU_CODENAME=resolute$
LOGO=ubuntu-logo$
```

\$ cat -n /etc/networks

```
1      # symbolic names for networks, see
networks(5) for more information
2      link-local 169.254.0.0
```

\$ cat /etc/networks /etc/os-release

```
# symbolic names for networks, see networks(5) for
more information
link-local 169.254.0.0
PRETTY_NAME="Ubuntu 26.04 LTS"
NAME="Ubuntu"
VERSION_ID="26.04"
VERSION="26.04 LTS (Resolute Raccoon)"
VERSION_CODENAME=resolute
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=resolute
LOGO=ubuntu-logo
```

\$ cat /etc/os-release

```
PRETTY_NAME="Ubuntu 26.04 LTS"
NAME="Ubuntu"
VERSION_ID="26.04"
VERSION="26.04 LTS (Resolute Raccoon)"
VERSION_CODENAME=resolute
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
```

```
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=resolute
LOGO=ubuntu-logo
```

\$ cat -n /etc/os-release /etc/networks

```
1      PRETTY_NAME="Ubuntu 26.04 LTS"
2      NAME="Ubuntu"
3      VERSION_ID="26.04"
4      VERSION="26.04 LTS (Resolute Raccoon)"
5      VERSION_CODENAME=resolute
6      ID=ubuntu
7      ID_LIKE=debian
8      HOME_URL="https://www.ubuntu.com/"
9      SUPPORT_URL="https://help.ubuntu.com/"
10
11      BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
12
13      PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
14      UBUNTU_CODENAME=resolute
15      LOGO=ubuntu-logo
16      # symbolic names for networks, see
17      networks(5) for more information
18      link-local 169.254.0.0
```

4. nslookup — query Internet name servers interactively

۴. nslookup

پرس و جوی تعاملی از سرورهای نام اینترنتی

```
$ whatis nslookup  
nslookup - query Internet name servers  
interactively
```

nslookup از DNS برای تبدیل یک نام هاست به آدرس IP آن (یا برعکس) استفاده می‌کند. هر سه مثال عملی دامنه‌های واقعی را در برابر resolver محلی systemd-resolved به آدرس 127.0.0.53 پرس و جو می‌کنند، که آدرس پیش‌فرض resolver در Ubuntu 26.04 LTS است.

```
$ nslookup www.mywebuniversity.com
```

```
Server:                127.0.0.53  
Address:               127.0.0.53#53  
  
Name:  www.mywebuniversity.com  
Address: 192.168.254.29
```

```
$ nslookup ourunix.com
```

```
Server:                127.0.0.53  
Address:               127.0.0.53#53  
  
Non-authoritative answer:  
Name:  ourunix.com  
Address: 47.180.19.94
```

```
$ nslookup -q=A www.mywebuniversity.com
```

```
Server:                127.0.0.53  
Address:               127.0.0.53#53  
  
Name:  www.mywebuniversity.com  
Address: 192.168.254.29
```

آدرس 53#127.0.0.53 شنونده stub محلی -systemd-resolved است، نه یک سرور DNS خارجی — آن پرس و جوها را به resolver های بالادستی تنظیم شده شما ارسال می کند (به /etc/resolv.conf و فصل ۴۳ برای معلومات بیشتر در مورد resolution ی DNS مراجعه کنید).

5. mkdir — make directories

```
$ whatis mkdir
```

```
mkdir (1) - make directories
```

`mkdir` یک یا چند دایرکتوری خالی جدید ایجاد می‌کند. برخلاف چهار کمنده قبلی، `mkdir` در صورت موفقیت هیچ خروجی تولید نمی‌کند — سکوت به معنای تأیید است، و شما نتیجه را با `ls` بعداً تأیید می‌کنید. هر دو مثال عملی از سایت همراه در ذیل نشان داده شده، با آن مرحله تأیید اضافه‌شده.

```
$ mkdir new
```

```
$ ls -ld new
```

```
drwxr-xr-x 2 wahid wahid 4096 Jul 25 09:00  
new
```

یک دایرکتوری جدید به نام `new` در دایرکتوری کاری فعلی ایجاد می‌کند.

```
$ mkdir -p dir1/subdir-A/subdir-B
```

```
$ mkdir -p dir1/subdir-A/subdir-B
```

```
$ find dir1
```

```
dir1
```

```
dir1/subdir-A
```

```
dir1/subdir-A/subdir-B
```

پرچم `-p` (parents) دایرکتوری `dir1` را ایجاد می‌کند، سپس `subdir-A` را درون آن، و سپس `subdir-B` را درون آن، همه در یک کمنده واحد — و مهم‌تر اینکه، اگر برخی از آن دایرکتوری‌های والد از قبل موجود باشند خطا نمی‌دهد، که آن را به گزینه پیش‌فرض امن‌تر برای اسکریپت‌هایی تبدیل می‌کند که نیاز دارند یک مسیر دایرکتوری تضمین‌شده وجود داشته باشد.

Chapter 3 Command Deep-Dives

فصل ۳: بررسی عمیق دستورات

chmod • chgrp • chown • passwd • alias

نویسبرداری و فارمت‌بندی‌شده از -MyWebUniversity.com/cgi-bin/CH3.cgi، مطابق با فصل ۳: کمندهای متوسط یونیکس و لینوکس.

A Note on UNIX and Linux Examples

یادداشتی درباره نمونه‌های یونیکس و لینوکس

اگرچه این کتاب *The Direct Path to Linux Ubuntu* نامیده می‌شود و عمدتاً بر اساس *Ubuntu 26.04 LTS* نوشته شده، متوجه خواهید شد که پنج نشست عملی این فصل روی *OpenSolaris* ثبت شده‌اند نه *Ubuntu* — که در جزئیاتی مانند *uid=101(wahid)*، گروه *staff*، و استفاده از *su* و *zoneadm* - قابل مشاهده است. این عمدی است، نه یک اشتباه. همان‌طور که فصل ۱ توضیح داد، لینوکس یک سیستم عامل شبیه به یونیکس است که به طور مستقل ساخته شده اما همان زیربنای *POSIX* را به اشتراک می‌گذارد، و کمندهایی مانند *alias*، *chmod*، *chgrp*، *chown*، *passwd* تقریباً به یک شکل در یونیکس و لینوکس رفتار می‌کنند دقیقاً به این دلیل که آن میراث مشترک عمیق است. آوردن مثال‌ها از هر دو سنت یک انتخاب آگاهانه است: این به شما تفاوت‌های کوچک و واقعی را نشان می‌دهد که به عنوان یک سیستم ادمین کاری هنگام جابه‌جایی بین پلتفرم‌ها — *Solaris*، *Ubuntu*، *Red Hat*، یا هر پلتفرم دیگر — واقعاً با آن‌ها روبرو خواهید شد، به‌جای اینکه وانمود کند هر سیستم خانواده یونیکس دقیقاً یکسان رفتار می‌کند. هر جایی که یک جزئیات مخصوص *Solaris* باشد (مانند *cmd* *zoneadm* یا رفتار محیط *su* -) نه یک چیز جهانی، این کتاب آن را به وضوح مشخص می‌کند تا همیشه بدانید کدام بخش‌ها معلومات قابل انتقال هستند و کدام‌ها مخصوص یک پلتفرم خاص.

1. chmod — change file mode bits

```
$ whatis chmod
```

```
chmod (1) - change file mode bits
```

```
chmod (1) - change the permissions mode of a file
```

```
chmod (2) - change access permission mode of file
```

chmod صلاحیت‌های دسترسی را برای یک یا چند فایل و دایرکتوری تغییر می‌دهد. برای درک کامل آن، ابتدا باید umask را بفهمید، که صلاحیت‌های پیش‌فرض فایل‌ها و دایرکتوری‌های جدید را تعیین می‌کند. در این نشست مثال، کاربر wahid یک umask با مقدار 0022 دارد، به این معنی که دایرکتوری‌های جدید به طور پیش‌فرض (755) rwxr-xr-x و فایل‌های جدید به طور پیش‌فرض (644) rw-r--r-- می‌شوند — مقدار umask از حداکثر صلاحیت ممکن (777) برای دایرکتوری‌ها، 666 برای فایل‌ها) کم می‌شود.

Permission bits, symbolic and octal

بیت‌های مجوز، نمادین و هشت‌هشتی (اکتال)

Symbolic نمادین	Octal هشتی	Meaning معنی
rwx	7	تمام صلاحیت‌ها: خواندن، نوشتن، و اجرا
rw-	6	خواندن و نوشتن، بدون اجرا
r-x	5	خواندن و اجرا، بدون نوشتن
r--	4	فقط خواندن
-wx	3	نوشتن و اجرا، بدون خواندن
-w-	2	فقط نوشتن

--x	1	فقط اجرا
---	0	هیچ دسترسی‌ای نیست

هر رقم مجموع خواندن (۴)، نوشتن (۲)، و اجرا (۱) برای
 owner، group و others به ترتیب است. برای مثال،
 خواندن+اجرا برای group و others — $rw\text{xr-xr-x} = 7,5,5 = 755$ دسترسی کامل برای owner،
 صلاحیت معمولی برای یک اسکریپت اجرایی.

Checking identity and umask

بررسی هویت

و umask

```
$ whoami
wahid
$ id
uid=101(wahid) gid=10(staff)
groups=10(staff)
$ umask
0022
```

ایجاد یک فایل و دایرکتوری، و خواندن صلاحیت‌های پیش‌فرض آن‌ها

```
$ mkdir newdir
$ touch newfile
$ ls -ld newdir newfile
drwxr-xr-x 2 wahid staff 2 2011-01-15 21:09
newdir
-rw-r--r-- 1 wahid staff 0 2011-01-15 21:10
newfile
```

$777 - 022 = 755$ برای دایرکتوری جدید؛ $666 - 022 = 644$ برای فایل جدید — دقیقاً همان‌طور که umask پیش‌بینی می‌کند.

Changing permissions with chmod

```
$ chmod 777 newdir
$ ls -ld newdir
drwxrwxrwx 2 wahid staff 2 2011-01-15 21:09
newdir

$ chmod 755 newdir
$ ls -ld newdir
drwxr-xr-x 2 wahid staff 2 2011-01-15 21:09
newdir

$ chmod 664 newfile pkg-publisher this
$ ls -l
total 16
drwxr-xr-x 2 wahid staff 2 2011-01-15 21:09
newdir
-rw-rw-r-- 1 wahid staff 0 2011-01-15 21:10
newfile
-rw-rw-r-- 1 wahid staff 11258 2011-01-03
22:09 pkg-publisher
drwxr-xr-x 2 wahid staff 8 2011-01-15 20:02
text
-rw-rw-r-- 1 wahid staff 0 2011-01-15 21:29
this
```

قابل اجرا ساختن یک اسکریپت

یک مورد واقعی و رایج: یک شل اسکریپت که با `echo` ایجاد شده بیت اجرا (execute bit) ندارد، پس اجرای مستقیم آن با پیام «Permission denied» ناکام می‌شود تا اینکه `chmod` بیت اجرا را اضافه کند.

```
$ echo "df -hF zfs" > df-zfs.sh
$ ls -l df-zfs.sh
-rw-r--r-- 1 wahid staff 11 2011-01-17 13:01
df-zfs.sh

$ ./df-zfs.sh
bash: ./df-zfs.sh: Permission denied
```

```
$ sh -x df-zfs.sh
```

Run without changing permissions, using sh's
-x trace mode

```
$ df -hF zfs
```

Filesystem	Size	Used	Avail
Use% Mounted on			
rpool/ROOT/opensolaris-1	20G	7.2G	12G
38% /			

```
$ chmod 755 df-zfs.sh
```

```
$ ls -l df-zfs.sh
```

```
-rwxr-xr-x 1 wahid staff 11 2011-01-17 13:01  
df-zfs.sh
```

```
$ ./df-zfs.sh # now runs directly,  
since the execute bit is set
```

Filesystem	Size	Used	Avail
Use% Mounted on			
rpool/ROOT/opensolaris-1	20G	7.2G	12G
38% /			

2. chgrp — change group ownership

chgrp:۲

— تغییر مالکیت گروهی

```
$ whatis chgrp
```

```
chgrp (1) - change group ownership
```

```
chgrp (1) - change file group ownership
```

chgrp گروهی که مالک یک فایل یا دایرکتوری است را تغییر می‌دهد. مانند chmod و chown، به صلاحیت کافی نیاز دارید — یا از قبل عضو گروه هدف باشید، یا دسترسی root داشته باشید — تا این تغییر را انجام دهید.

Checking current group ownership

بررسی مالکیت فعلی گروه

```
$ pwd
```

```
/export/home/wahid/training/OpenSolaris/pkgd  
ir
```

```
$ ls -l
```

```
total 13
```

```
-rw-r--r-- 1 wahid staff 11258 2011-01-03
```

```
22:09 pkg-publisher
```

```
drwxr-xr-x 2 wahid staff 8 2011-01-15 17:36
```

```
text
```

یک تلاش بدون صلاحیت همان‌طور که انتظار می‌رفت ناکام می‌شود

```
$ id
```

```
uid=101(wahid) gid=10(staff)
```

```
groups=10(staff)
```

```
$ chgrp root *
```

```
chgrp: changing group of `pkg-publisher':
```

```
Not owner
```

```
chgrp: changing group of `text': Not owner
```

wahid فقط عضو گروپ **staff** است، بنابراین تلاش برای واگذاری مالکیت به گروپ **root** رد می‌شود — شما نمی‌توانید یک فایل را به گروپی که عضو آن نیستید بدون صلاحیت بالاتر بدهید.

Switching to root and changing group ownership

تغییر وضعیت به کاربر ریشه

root

و تغییر مالکیت گروهی

```
$ su -
Password:
# cd
/export/home/wahid/training/OpenSolaris/pkgdir
# chgrp root *
# ls -l
total 13
-rw-r--r-- 1 wahid root 11258 2011-01-03
22:09 pkg-publisher
drwxr-xr-x 2 wahid root 8 2011-01-15 17:36
text
```

توجه کنید که **chgrp** بدون **-R** بازگشتی (recursive) نیست: خود دایرکتوری **text** به گروپ **root** تغییر کرد، اما فایل‌های درون آن تغییر نکردند.

Recursive group change with -R

تغییر گروه بازگشتی با

— **R**

```
# chgrp -R root *
# ls -lR text
```



```
text:
-rw-r--r-- 1 wahid root 2856 2011-01-03
22:04 pkg-info.txt
-rw-r--r-- 1 wahid root 475569 2011-01-03
23:38 pkg-install-amp.txt
-rw-r--r-- 1 wahid root 2358 2011-01-03
22:06 pkg-list.txt

# chgrp -R staff *           # revert back to
the original group
# ls -l
-rw-r--r-- 1 wahid staff 11258 2011-01-03
22:09 pkg-publisher
drwxr-xr-x 2 wahid staff 8 2011-01-15 17:36
text
```

chown — change file owner and group

chown

— تغییر مالک و گروه فایل

```
$ whatis chown
```

```
chown (1) - change file owner and group
```

```
chown (1) - change file ownership
```

```
chown (2) - change owner and group of a file
```

chown کاربر مالک یک فایل یا دایرکتوری را تغییر می‌دهد، و به صورت اختیاری، همزمان گروپ را نیز با سینتکس owner:group تغییر می‌دهد. مانند chgrp، تغییر مالکیت معمولاً به صلاحیت root نیاز دارد.

یک تلاش بدون صلاحیت ناکام می‌شود

```
$ pwd
```

```
/export/home/wahid/training/OpenSolaris/pkgdir
```

```
$ id
```

```
uid=101(wahid) gid=10(staff)
```

```
groups=10(staff)
```

```
$ ls -l
```

```
-rw-r--r-- 1 wahid staff 11258 2011-01-03
```

```
22:09 pkg-publisher
```

```
drwxr-xr-x 2 wahid root 8 2011-01-15 20:02
```

```
text
```

```
$ chown root text
```

```
chown: changing ownership of `text': Not owner
```

به عنوان root: تغییر فقط مالک

```
$ su
```

```
Password:
```

```
# pwd
```

```
/export/home/wahid/training/OpenSolaris/pkgd
ir
# chown root text
# ls -l
-rw-r--r-- 1 wahid staff 11258 Jan  3 22:09
pkg-publisher
drwxr-xr-x 2 root  root  8 Jan 15 20:02 text
```

تغییر مالک و گروپ همزمان

```
# chown wahid:staff text
# ls -l text
-rw-r--r-- 1 wahid staff  2856 Jan  3 22:04
pkg-info.txt
-rw-r--r-- 1 wahid staff 475569 Jan  3 23:38
pkg-install-amp.txt
```

تغییر بازگشتی مالکیت با -R

```
# chown -R root:sys text
2 # ls -ld text
drwxr-xr-x 2 root sys 8 Jan 15 20:02 text
# ls -l text
-rw-r--r-- 1 root sys  2856 Jan  3 22:04
pkg-info.txt
-rw-r--r-- 1 root sys 475569 Jan  3 23:38
pkg-install-amp.txt

# chown -R wahid text          # restore
ownership
# exit
$ id
uid=101(wahid) gid=10(staff)
groups=10(staff)
```

مانند `chgrp`، همیشه قبل و بعد از هر تغییر مالکیت با `ls -l` (یا `ls -ld` برای خود دایرکتوری، بدون وارد شدن به آن) دوباره بررسی کنید — یک `chown -R` تصادفی روی دایرکتوری اشتباه می‌تواند شما را از فایل‌های خودتان محروم کند.

4. passwd — change login password and password attributes

passwd .۴

— تغییر رمز عبور ورود و ویژگی‌های رمز عبور

```
$ whatis passwd  
passwd - change login password and password  
attributes
```

passwd پسورد لاگین یک کاربر را تغییر می‌دهد. هر کاربر می‌تواند پسورد خودش را صرف با اجرای passwd بدون هیچ آرگومنتی تغییر دهد؛ تغییر پسورد کاربر دیگر، یا پسورد root، به صلاحیت root نیاز دارد.

Changing another user's password as root

تغییر رمز عبور کاربر دیگر با دسترسی روت

(root)

```
# id  
uid=0(root) gid=0(root)  
groups=0(root),1(other),2(bin),3(sys)...
```

```
# passwd -r files devuser  
New Password:  
Re-enter new Password:  
passwd: password successfully changed for  
devuser
```

آپشن r files- مشخص می‌کند که تغییر باید در فایل محلی directory/etc/shadow/ نوشته شود، نه یک سرویس directory شبکه‌ای — در سیستم‌هایی که از تصدیق هویت متمرکز استفاده می‌کنند، r ldap- یا r nis- به جای آن LDAP یا NIS را هدف قرار می‌دهد، یک تفاوت مهم که باید بفهمید وقتی به پوشش فصل ۳۶ در مورد LDAP و directory services برسید.

Preventing password expiration

جلوگیری از انقضای رمز عبور

```
# passwd -x -1 devuser
```

```
passwd: password information changed for  
devuser
```

آپشن -x -1 حداکثر عمر پسورد را غیرفعال می‌کند، پس حساب هرگز به طور خودکار مجبور به تغییر پسوردش نمی‌شود — این برای حساب‌های سرویس مفید است، هرچند طبق بیشتر پالیسی‌های امنیتی برای کاربران انسانی تعاملی توصیه نمی‌شود.

Verifying the change

تأیید تغییر

```
# grep -i devuser /etc/passwd
```

```
devuser:x:101:10:WahidLutfy:/home/devuser:/b  
in/bash
```

```
# grep -i devuser /etc/shadow
```

```
devuser:$6$illustrativeHashOnly$doNotUseInPr  
oduction:15061:::::::
```

توجه: خط `/etc/shadow/` در بالا با یک هش نمایشی جایگزین نشان داده شده نه یک هش واقعی — هرگز یک هش پسورد واقعی را منتشر نکنید، حتی از یک محیط آموزشی، زیرا ابزارهای کرکینگ آفلاین گاهی می‌توانند پسورد اصلی را از آن بازیابی کنند.

5. alias — create a shortcut name for a command

۵. alias

— ایجاد یک نام میان‌بر برای یک دستور

```
$ whatis alias
```

A shorter name for assigning UNIX commands, scripts, and other programs.

alias به شما اجازه می‌دهد یک نام کوتاه و به‌یادماندنی برای یک کمنند طولانی‌تر یا یک pipeline از کمندها تعریف کنید. سینتکس دقیق آن بستگی به این دارد که کدام شل استفاده می‌کنید: شل‌های خانواده Bourne (sh، ksh، bash) از `'name'=command` استفاده می‌کنند، در حالی که شل‌های خانواده C (csh، tcsh) از `name command` بدون علامت مساوی و بدون نیاز به علامت نقل‌قول اطراف کمندهای ساده استفاده می‌کنند.

Bourne/Bash syntax

Bourne/Bash سینتکس

```
$ ps
```

```
PID TTY  TIME CMD
707 pts/3 0:03 bash
```

```
$ alias myf='find . -name "*" -print -exec
ls -l {} \;'
$ alias myls='ls -ltr'
```

```
$ alias
```

```
alias myf='find . -name "*" -print -exec ls
-l {} \;'
```

```
$ alias myls='ls -ltr'
```

alias می‌تواند هر کمندی را با هر آپشنی که از قبل در آن گنجانده شده، بپیچاند — از جمله یک مثال مخصوص ادمین برای سیستمی که مدیریت می‌شود:

```
$ alias myz1='zoneadm list'
```

```
$ myz1  
global
```

```
$ alias myz1='zoneadm list -p' #  
redefine with an added option  
$ myz1  
0:global:running:/::native:shared
```

اجرای alias به نام myf (که معادل یک ls -lR بازگشتی است) تأیید می‌کند که alias دقیقاً مانند تایپ کردن تعریف کامل خودش رفتار می‌کند:

```
$ myf  
.  
total 1  
-rw-r--r-- 1 wahid staff 0 2011-01-09 17:56  
newfile  
./newfile  
-rw-r--r-- 1 wahid staff 0 2011-01-09 17:56  
./newfile
```

C shell (csh/tcsh) syntax

سینتکس C shell (csh/tcsh)

تغییر شل هم پرامپت (به %) و هم سینتکس alias را تغییر می‌دهد — بدون علامت مساوی، و کمند معمولاً برای موارد ساده بدون علامت نقل قول باقی می‌ماند:

```
$ /bin/csh  
% ps  
PID TTY TIME CMD  
1585 pts/3 0:00 csh
```



```
% alias myz zoneadm list -cv
% alias mydf df -hF ufs
% alias
myz (zoneadm list -cv)
mydf (df -hF ufs)
```

alias ها به طور پیش فرض در هر دو خانواده شل فقط برای همان نشست (session-local) هستند — وقتی شل بسته می شود، ناپدید می شوند. برای دائمی ساختن یک alias، همان تعریف alias را به فایل startup شل خود اضافه کنید: bashrc./~ برای bash، یا cshrc./~ برای csh/tcsh، و یک شل جدید باز کنید (یا فایل را source کنید) تا فعال شود.

Continue the Full Edition

All 43 Chapters — Available Now in eBook, Paperback,
and Hardcover

شما همین حالا **Preface** و فصل‌های ۱ تا ۳ کتاب **The Direct Path to Linux Ubuntu** را خواندید. نسخه کامل ۴۳ فصلی با این‌ها ادامه می‌یابد:

- Chapter 4 — Advanced UNIX & Linux Commands (user/group management, systemctl, tar/gzip, systemd, dpkg, apt, lsof)
- Chapter 5 — Shell Scripting (Bash & C-shell scripts, conditionals, and the LUBE practice terminal)
- Chapter 6 — UNIX and Linux Web Servers (Apache fundamentals)
- Chapter 7 — The Dynamic Linux Man Page Lookup Tool
- Chapter 8 — The PDF Man Page Library, Browsing by Section
- Chapter 9 — The C and C++ Include Header Files
- Chapter 10 — The Rocky Linux 9.2 Manual Pages
- Chapter 11 — The Red Hat Enterprise Linux 9.2 Manual Pages
- Chapter 12 — Dynamic Generation of Linux Ubuntu Manual Pages (man2pdf)
- Chapter 13 — The Linux Ubuntu 22.04 LTS Manual Pages, Static Library
- Chapter 14 — Linux BASH and Microsoft PowerShell Integration
- Chapter 15 — Learn Linux Ubuntu Online, the LUBE Practice Terminal

- Chapter 16 — The Linux Ubuntu 26.04 Manual Pages in PDF Format
- Chapters 17–20 — Video Companion Resources (Long Lectures & Quick Tips, English and Dari)
- Chapter 21 — Linux Man Pages Online, PDF & HTML by Category
- Chapters 22–27 — C, C++, Go, Java, Ruby, and Rust
- Chapters 28–30 — Perl, R, JavaScript and Node.js
- Chapters 31–33 — HTML5/CSS3, JSON, and XML
- Chapters 34–35 — YAML and SQL Databases
- Chapters 36–37 — LDAP & Directory Services, and Cloud Computing
- Chapters 38–40 — Ansible & OpenShift, AWS & Azure, and Docker
- Chapters 41–43 — Apache & NGINX, Protocols/APIs & Security, and Networking
- Plus: Book-Wide Explanatory Additions and the full Index

Get the Full Edition

eBook

\$9.99

قیمت یکسان در هر دو پلتفرم

[Buy on Amazon](#)

|

[Buy on Barnes & Noble](#)

Paperback

\$24.99–\$34.99

قیمت بسته به پلتفرم متفاوت است

[Buy on Amazon](#)

|

[Buy on Barnes & Noble](#)

Hardcover

\$44.99

قیمت یکسان در هر دو پلتفرم

[Buy on Amazon](#)

|

[Buy on Barnes & Noble](#)

از اینکه این کتاب را خواندید متشکریم. اگر این **Free Starter Edition** به شما کمک کرد، لطفاً برای درس‌های ویدیویی رایگانی که با هر فصل همراه است، سبسکرایب کنید:

[YouTube.com/@MyWebUniversity](https://www.youtube.com/@MyWebUniversity)

(انگلیسی) و

[YouTube.com/@MyWebUniversity-Dari](https://www.youtube.com/@MyWebUniversity-Dari)

(دري).